A "Virtual Unix" for Tops-20

The major elements involved in providing a Unix environment under Tops-20 are reported. The project's background and evolution are explained, including the marked contrast between our original motivation and goals and the current rationale. Not only are there distinct problems involved in adapting the so-called "portable" C compiler to a word-addressed machine, but the C language "specification" itself often proves ambiguous when the target machine is dissimilar to the canonical pdp-11 architecture. The "natural" choice for byte size on the Dec-20 of 7 bits is eliminated by the compiler, the C spec, and particularly the Unix i/o routines. The ways in which my problems and solutions differed from Sam Leffler's work with pcc on the Harris/6, another word-addressed machine, are described. As for quality of code, the use of a powerful re-targetable peephole optimiser which accepts an ISP machine description [1] can significantly simplify the compiler itself.

"Virtual Unices" have been layered on other machines and operating systems, and in most instances they simply provide a separate, and superior, environment. It is not so simple in Tops-20's case, however, for that operating system offers several features that are attractive even to the Unix purist, and many users are loathe to forsake these niceties. This makes the smooth integration of Unix and Tops-20 facilities difficult; indeed, it might well have been easier to bring up Unix itself than to build a run-time-system which allowed access to both environments. Section 2 of the Unix programmers' manual comes close to defining Unix itself, but there are some common programming practices which assume more. A few system calls proved difficult or impossible to emulate, but the immense payoff from succeeding in that approach clearly outweighed the short-term gains obtainable by aiming for Section 3 compatibility only. Finally, I review our record in porting large V7 user programs, such as make, sh, and uucp, and give some performance figures.

1. The Design and Application of a Retargetable Peephole Optimiser, J.W. Davidson & C.W. Fraser, TOPLAS, 2, 2 (April 1980), 191-202.

Author:
Jay Lepreau
Computer Science Dept.
3160 MEB
University of Utah
Salt Lake City, UT 84112

Arpa: Lepreau@Utah-20
Uucp: decvax!randvax!utah-cs!lepreau

(801) 581-8322 -- Some afternoons
" 582-7214 -- Most mornings, some afternoons.
" 581-8224 -- Can leave messages, 8-5 MST

Audio-visual requirements:
Overhead projector